Recommending Personalized Summaries of Teaching Materials

IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received January 20, 2019, accepted February 8, 2019, date of publication February 15, 2019, date of current version March 4, 2019.

Digital Object Identifier 10.1109/ACCESS.2019.2899655

Recommending Personalized Summaries of Teaching Materials

LUCA CAGLIERO, (Member, IEEE), LAURA FARINETTI, (Member, IEEE), AND ELENA BARALIS, (Member, IEEE)
Dipartimento di Automatica e Informatica, Politecnico di Torino, 24 10129 Turin, Italy
Corresponding author: Luca Cagliero (luca.cagliero@polito.it)

ABSTRACT Teaching activities have nowadays been supported by a variety of electronic devices. Formative assessment tools allow teachers to evaluate the level of understanding of learners during frontal lessons and to tailor the next teaching activities accordingly. Despite plenty of teaching materials are available in the textual form, manually exploring these very large collections of documents can be extremely time-consuming. The analysis of learner-produced data (e.g., test outcomes) can be exploited to recommend short extracts of teaching documents based on the actual learner's needs. This paper proposes a new methodology to recommend summaries of potentially large teaching documents. Summary recommendations are customized to student's needs according to the results of comprehension tests performed at the end of frontal lectures. Specifically, students undergo multiple-choice tests through a mobile application. In parallel, a set of topic-specific summaries of the teaching documents is generated. They consist of the most significant sentences related to a specific topic. According to the results of the tests, summaries are personally recommended to students. We assessed the applicability of the proposed approach in real context, i.e., a B.S. university-level course. The results achieved in the experimental evaluation confirmed its usability.

INDEX TERMS Learning analytics, personalized summary recommendation, text summarization.

I. INTRODUCTION

In recent years the diffusion of e-learning platforms has radically changed the way of transferring knowledge. Teachers can easily share teaching materials, such as electronic books, slides, scientific papers, lecture notes, videos, images, with learners through Web-based or mobile applications. The advent of electronic devices has simplified the interaction between teachers and learners [1]. For instance, learners can easily access online teaching materials, submit assignments or reports remotely, and undergo assessment tests during or after the lectures.

Formative assessment tools [2]–[4] are learning applications focused on monitoring students' progress by providing teachers ongoing feedback. The aim of these tools is twofold. Firstly, they help learners to identify their weaknesses and strengths thus targeting areas in which they need to deepen their knowledge. Secondly, they support teachers in recognizing where students are struggling, thus allowing them to promptly overcome learning issues. To gain insights into the learning process, the outcomes of formative assessment tests can be collected and analyzed. Learning analytics tools analyze learner-produced data in order to support teaching activities. [5].

This work focuses on analyzing textual data, which represent the most widespread learning content type [6]. Specifically, we consider textual documents (e.g., teaching books, scientific paper, learning notes) as reference teaching materials. As learner-produced data, we analyze the content and outcomes of multiple-choice tests written in textual form (i.e., we disregard highlights, images, video, or other multimedia content).

Since the amount of learner-generated and teaching materials available in textual form is increasing, manually exploring these materials is often practically unfeasible. To simplify the exploration of teaching documents, we propose to extract short summaries consisting of the most significant sentences pertinent to specific topics. Previous research works have already highlighted the usefulness of short textual summaries to support learning activities [7]. Specifically, exploring automatically generated summaries in place of the original documents (i) expedites the preliminary exploration of verbose documents, (ii) simplifies the review of previously studied content, and (iii) improves content accessibility in learning

The associate editor coordinating the review of this manuscript and approving it for publication was Jerry Chun-Wei Lin.

contexts in which the adopted devices have either limited bandwidth or low resolution.

This paper proposes a new methodology, namely Test-driven Summarization (TestSumm, in short), to recommend topic-specific summaries to learners based on their individual needs. At the end of frontal lectures, learners undergo multiple-choice tests through a mobile application to assess their comprehension level of the given lesson. TestSumm considers the content and the outcomes of multiple-choice tests, performed at the end of frontal lessons, to automatically generate and recommend topic-specific summaries of the teaching documents. Specifically,

- The content of the tests is exploited to drive the generation of topic-specific summaries. Summary extraction relies on an ad hoc itemset-based strategy, which extends the state-of-the-art summarization algorithm proposed by [8]. The output summaries consist of the subset of sentences that are most relevant for specific topics.
- The test outcomes are used to personalize summary recommendation based on the actual learner's needs. Depending on the learner's outcomes, summary recommendations can be targeted to broad subjects covered by many questions or to very specific topics covered by single questions or answers.

The sentences included in the summary are linked to the original documents to simplify the retrieval of the original document content. Learners who are interested in deepening their knowledge on some specific aspects may follow the links to the original documents.

We assessed the usability of the proposed methodology in the context of a B.S. course of our university. Specifically, at the end of the lessons of the course, we performed multiple-choice tests. Based on the tests' outcomes, we identified the topics that each student may need to revise. In parallel, we extracted short textual summaries pertinent to different topics from the reference course textbook. Finally, we simulated the summary recommendation process and we evaluated the pertinence of the generated summaries by comparing each recommended sentence with a ground truth provided by the teacher of the course. The automatically generated summaries are, to a large extent, pertinent to the ground truth (i.e., 73% of the sentences appearing in the top five sentences of the summary were shortlisted by the domain expert).

The organization of the paper is described below. Section II overviews the related literature. Section III thoroughly describes the methodology. Section IV summarizes the main experimental results. Finally, Sections V and VI present future developments and draw conclusions, respectively.

II. RELATED WORK

Learning data are typically extensive and heterogeneous [5]. They may include textual documents (e.g. books, textual notes), multimedia content (e.g. videos, slides, images, charts), learner-produced data (tests, surveys, annotations), or other related content (e.g. highlights, lecture notes, grades). A large body of work has been devoted to capturing interesting patterns from learning data by means of data mining techniques [9]. As an example, supervised techniques, such as classification and regression, have been exploited to predict students' performance [10], [11] or instructor's performance [12]. Offering improved or personalized learning services is another remarkable research direction [13]–[16].

In [17]–[20] summarization techniques have been applied in the learning context. The work presented by [20] focused on automatically answering specific learners' questions. Unlike [20], in the methodology presented in this paper the actual learners' needs are automatically inferred from the content and outcomes of formative assessment tests. In [17] summaries of scientific articles are generated by adapting their content to the current knowledge level of the users. However, since assigning a priori knowledge levels to a learner can be challenging, our methodology proposes to integrate formative assessment strategies upstream to recommend summaries tailored to specific learners' needs. Therefore, the analyzed problem is complementary to those addressed by this work. A preliminary attempt to recommend summaries of teaching documents based on topic dictionaries associated with multiple-choice questions has been presented by [18]. This work extends the aforesaid study to a large extent. Specifically, its innovative contributions can be summarized as follows:

(i) It presents and thoroughly describes a new methodology to automatically recommend summaries tailored to learners' needs.

(ii) It proposes to drive summary generation by using not only a fixed topic dictionary but also single questions (question summaries) or single answers (answer questions). Question/answer summaries allow teachers to give punctual clarifications on specific concepts in case recommendations of general on-topic summaries are deemed as not appropriate for teaching purposes.

(iii) It quantitatively evaluates summarizer performance on real learner-generated data acquired in a real learning context.

(iv) It studies the setting of the algorithm configuration.

(v) It discusses the open challenges and the future research directions in the context of summarization of learning materials.

A parallel branch of research has been focused on proposing new document summarization algorithms. Depending on the type of generated summaries, two main approaches to text summarization have been proposed. *Sentence-based* approaches entail partitioning the document(s) into sentences and selecting the most informative ones to be included in the summary [8], [21]–[23]. Conversely, *keyword-based* approaches identify keywords to summarize the document content [24], [25]. Existing summarization approaches produce either general-purpose summaries (i.e., summaries that are not tailored to any specific topic), or topic-specific summaries tailored to a (analyst-provided) domain-specific dictionary. In our context, existing solutions cannot be directly integrated and used in our methodology, because

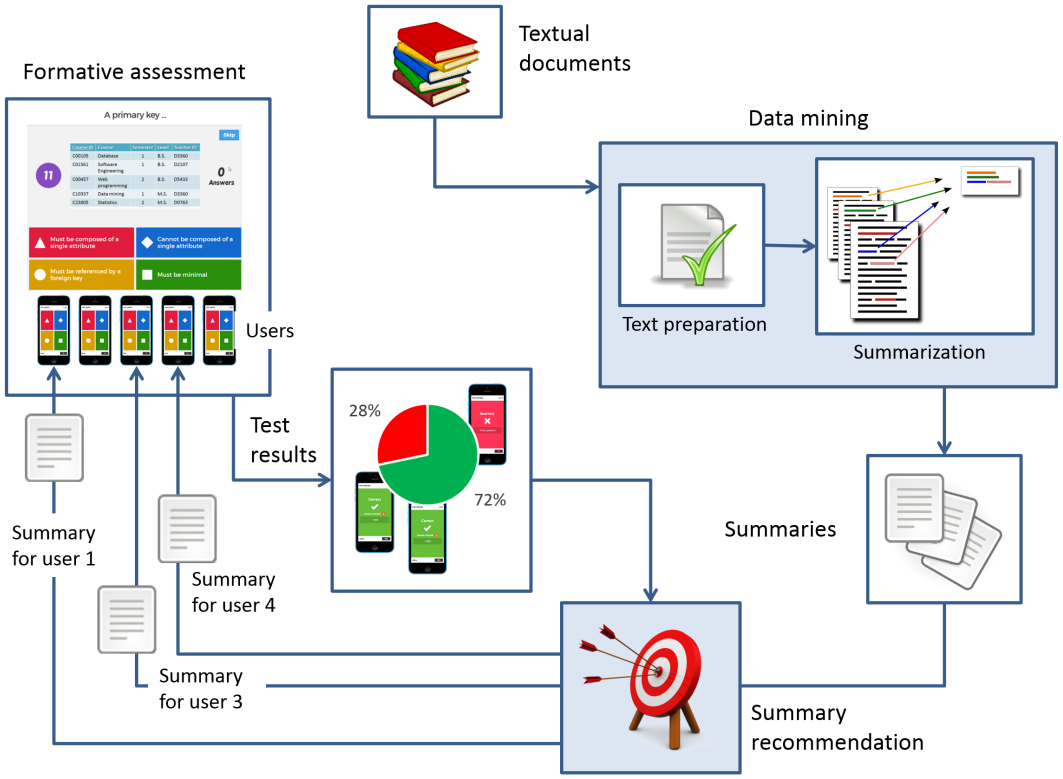

FIGURE 1. The TEST-driven SUMmarization architecture.

the domain-specific dictionary is not given a priori, but it must be inferred from the content of the multiple-choice tests. Furthermore, based on the tests' outcomes, summaries at different level of detail should be generated and recommended. This work extends an itemset-based summarizer [8] by integrating topic-specific information derived from multiple-choice tests.

III. THE TEST-DRIVEN SUMMARIZATION METHODOLOGY

This section presents the proposed methodology, whose main steps are depicted in Figure 1.

- *Formative assessment.* For each frontal lecture, it assesses the learner's level of understanding of different topics through multiple-choice tests (see Section III-A).
- *Text preparation.* It processes the text contained in multiple-choice tests and teaching materials to make it suitable for subsequent analyses (see Section III-B).
- *Summarization.* It summarizes the teaching materials by extracting the most significant content pertinent to each topic (see Section III-C).
- *Summary recommendation.* It recommends ad hoc summaries generated from the teaching materials to learners who fail specific tests (see Section III-D).

A. FORMATIVE ASSESSMENT

The goal of this step is to assess learner's level of understanding during or immediately after frontal lectures. Getting early feedback on their level of comprehension allows teachers to monitor learner's progress and to tailor the next teaching activities accordingly.

Plenty of mobile or Web-based applications have already been proposed to support the assessment process [26]. Applications must be (i) easy-to-use (possibly self-explanatory), (ii) easily portable to different devices and operating systems, and (iii) interactive, to enable interactions between learners and teachers.

We implemented a draft prototype of the presented methodology, where we integrated Kahoot! formative assessment mobile application. During the test, learners are gathered around a common screen and equipped with an electronic device (e.g. smartphone or tablet). The test consists of a set of multiple-choice tests, with four possible answers displayed on the main screen. Learners choose the right answer to each question within a limited amount of time (typically, a few seconds). Multiple-choice tests are considered because: (i) they are quite simple and widely used, (ii) questions and answers typically contain key terms recalling the most salient topics [1]. In the following steps, the text appearing in the tests' questions and answers is analyzed in order to map learning summaries to tests.

B. TEXT PREPARATION

Let $\mathbf{D}$ be the set of textual documents $d_1, d_2, \ldots, d_n$ considered in our analyses. In this study we disregard non-textual content such as pictures and references in textbooks, slides, videos, highlights, and annotations.

Each document can be modeled as a set of sentences (i.e., portions of text separated by periods, question marks, or exclamation marks). Let s_j^i be the j-th sentence of document $d_i \in \mathbf{D}$. The goal is to generate a summary consisting of a selection of sentences s_j^i in $\mathbf{D}$.

To apply the summarization process to the document set, documents are transformed by exploiting the following established preprocessing steps [27].

- *Stopword elimination*: This step entails pruning the words with little semantic relevance (e.g., prepositions, conjunctions), because their occurrences in the text are not important for evaluating sentence relevance.
- *Stemming*: This step reduces words to their base form, otherwise the same word with a different inflection is treated as a different one.

Note that stopword elimination and stemming algorithms are currently available for a large variety of languages. Hence, the proposed methodology is portable to documents written in different languages.

The output of the text preparation phase is a document set $\mathbf{D}_p$, where each sentence is a bag-of-word (BOW), i.e., a set of word stems [27].

To select on-topic sentences, we will exploit a dictionary to drive the summarization process. Let $\mathbf{T}$ be the set of multiple choice tests. For each test $t_z \in \mathbf{T}$ let diz(t_z) be a dictionary consisting of all the stems that occur in the corresponding questions or answers (as discussed later, the selected content depends on the summary goal).

To each stem in the BOW of the document set $\mathbf{D}_p$ we assign a relevance score, which is a variant of the term frequency-document frequency (tf-df) statistics [8]. It considers three main factors:

(i) The frequency of the stem in each document (hereafter denoted as term frequency).

(ii) The number of documents in which the stem occurs at least once (denoted as document frequency).

(iii) The presence/absence of the stem in the dictionary (denoted as term rewarding/penalty score).

Specifically, the relevance score rs_{zi} of stem s_{zi} is computed as follows:

$$rs_{zi} = \Delta(s_{zi}) \cdot \frac{o_{zi}}{|d_i|} \cdot \frac{|\{d_i \in \mathbf{D}_p : s_{zi} \in d_i\}|}{|\mathbf{D}_p|}$$

where o_{zi} is the number of occurrences of the z-th stem s_{zi} in the i-th document d_i, $\mathbf{D}_p$ is the document set under analysis, $|d_i|$ is the number of stems that are contained in the i-th document d_i, $\frac{|\{d_i \in \mathbf{D}_p : s_{zi} \in d_i\}|}{|\mathbf{D}_p|}$ represents the document frequency of the stem s_{zi} in the whole document set, and $\Delta(s_{zi})$ is a function that returns a user-specified penalty score $\delta \in [0, 1]$ if stem s_{zi} is not present in the dictionary or 1 (no penalty) otherwise.

Since all the documents in the analyzed set are assumed to cover the same subject, we exploit a relevance score that gives higher importance to word stems that frequently occur both locally (within a document) and globally (in the document set), as they are deemed as the best representatives of the document content. To tailor summaries to the content of the tests, we reward word stems occurring in the dictionary, while penalizing the others. Th penalty score δ is set by the domain expert and may vary in the range [0,1]. The summaries generated by setting low δ values are more focused on the dictionary content, because the influence of the terms not appearing in the dictionary is less significant. If δ is set to zero only the word stems in the dictionary get non-zero relevance scores.

C. SUMMARIZATION

Given a test t_z in $\mathbf{T}$ and the dictionary diz(t_z) populated from t_z, this step focuses on extracting a summary $S(t_i)$ of the document set $\mathbf{D}$ pertinent to the test. To this purpose, sentence extraction is driven by the dictionary content.

We implemented the dictionary-driven summarization process by extending a state-of-the-art algorithm, i.e., the Multilingual Weighted Itemset-based Summarizer (MWISum) [8]. The MWISum summarizer relies on the following steps: (i) frequent itemset mining and (ii) sentence selection and ranking. The key idea behind the algorithm is to pick the sentences covering the largest number of combinations of frequently co-occurring word stems. A thorough description of each step is given below.

1) FREQUENT ITEMSET MINING

Itemset mining is a popular data mining technique used to describe data by means of recurrent patterns [28]. In the summarization algorithm [8], itemsets are exploited to represent sets of words with relatively high frequency of occurrence. To tailor summaries to the topic of the tests, we extended the original version of the summarizer provided by the authors by integrating the newly proposed relevance score described in Section III-B. In such a way, sentence evaluation and selection are driven by the dictionary content.

2) SENTENCE SELECTION AND RANKING

In this step, a subset of sentences are selected and included in the output summary. A sentence *covers* an itemset if it contains the corresponding combination of word stems. Since itemsets represent the most significant underlying correlations among words, the number of covered itemsets per sentence is exploited as the evaluation criterion of sentence relevance in the document set.

To generate a summary consisting of the most salient document content, the sentences that cover the largest number of weighted frequent itemsets are iteratively selected until all the itemsets in the model are covered by at least one sentence. The order of appearance of the sentences in the summary reflects their relative importance, i.e., sentences covering the largest number of itemsets are picked first to be included in the output summary.

D. SUMMARY RECOMMENDATION

This step entails generating personalized summary recommendation to learners who fail the tests. Figure 2 shows the

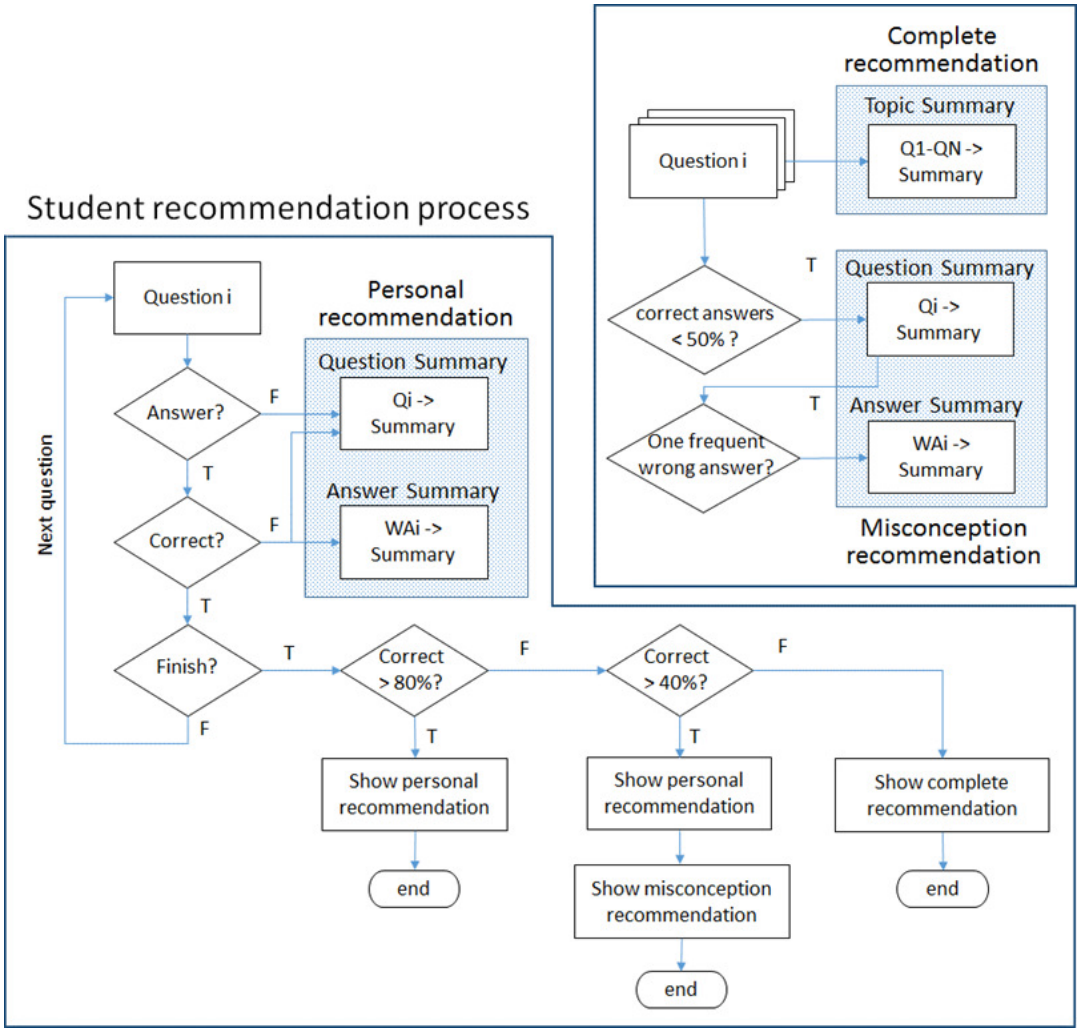

FIGURE 2. The recommendation process.

general strategy for providing personalized recommendation to learners. The picture is relative to a single topic (e.g., the *relational model* in a database course). Please note that in our experimentation the test is given to the students right after a lecture to assess their level of understanding, and not to verify retention of previously explained concepts. Each tests includes a set of questions, named Q_i in Figure 2. The summarization process can be driven by three different starting sets, according to the objective of the recommendation.

- The text of the whole set of questions, to get a general summary of the topic; this summary is named *topic summary* in the picture.
- The text of a specific question, to get a more specific summary related to a sub-area of the general topic; this summary is named *question summary* in the picture.
- The text of a specific (wrong) answer of a specific question, to get a more specific summary about a misconception; this summary is named *answer summary* in the picture.

Every time the student does not answer a question within the maximum allocated time, the system generates a *question summary* driven by that specific question text (Q_i in the picture). Every time he or she selects a wrong answer to a question, the system generates a *question summary* driven by the question text and an *answer summary* driven by the text of the selected wrong answer (WA_i in the picture). The automatically generated summaries constitute a *personalized recommendation* for the student. We decided to add the answer summary because wrong answers in this situation generally are a symptom of a concept misunderstanding. If student performance is good (i.e. he or she correctly answered to more than 80% of the proposed questions), we consider enough to deliver her or his personalized recommendation summary. If the student performance is average (i.e. he or she answered correctly to less than 80% of the proposed questions, but more than 40%), together with his or her personal recommendation summary, we deliver the *misconception recommendation* summary. Misconception recommendation relies on the evaluation of test performance of all the students, with

TABLE 1. Characteristics of multiple-choice tests.

Test Id	Num. of questions	Topics	Dictionary
1	8	Relational model	*Relation, Cardinality, Primary, Foreign, Key, Integrity, Constraint, Tuple, Domain, Uniqueness, Attribute, Record, Schema, Instance, Reference*
2	8	Transactions management	*Transaction, Commit, Rollback, Consistency, Durability, Atomicity, Reliability, ACID, State, Failure, Start, Automatic*
3	17	DB design	*Entity, Association, Cardinality, Attribute, Composite, Hierarchy, Transaction, Inheritance, Conceptual, Design, Normal, BCNF, Schema, Relationship*
4	22	SQL language	*Declarative, Language, Relation, Join, Instruction, Clause, Table, DBMS, Definition, Command, Select, Where, From, Join, Check, Like, Instruction, Clause, Operator, Not, Unique, Union, Create, Order*
5	18	Web programming	*DBMS, Client, Server, Application, Architecture, Tier, API, Call, Interface, Connection, JDBC, Layer*

the objective to identify the most difficult questions and the most frequent misconceptions. Every time more than 50% of the students wrongly answer to a given question, a *question summary* driven by that question text is generated; for the same question, in case one of the wrong answers has a number of selections higher than the correct one, an *answer summary* driven by the text of that wrong answer is generated. The collection of these summaries constitutes the *misconception recommendation*. Finally, if the student performance is insufficient (i.e. he or she correctly answered to less than 40% of the proposed questions), we deem that the best strategy is to recommend him or her a complete overview of the explained concepts. This is done by delivering the *topic summary*, without considering his or her specific mistakes, which likely cover a broad range of the topic.

In any case, delivering short personalized summaries, instead of the entire set of learning documents, allows students to overcome misconceptions in a more effective way. Furthermore, the direct links connecting summary and document content allow learners to deepen their knowledge on specific aspects which are not described in detail in the summary. Textual summaries are easy to share and visualize on mobile devices with limited computational power and network bandwidth. The experimental results, reported in the next section, demonstrated that, using our prototype of the proposed methodology, personalized summaries can be generated offline from the course textbooks in few seconds. A smarter implementation of the proposed methodology could integrate the summarization and recommendation steps into the formative assessment application.

IV. CASE STUDY

To assess the usability of the proposed methodology, we performed an evaluation experience during a university-level Bachelor of Science course on databases. At the end of five lectures, students were invited to login to the Kahoot! mobile app with a nickname and to undergo a test in anonymous form. Each test consisted of a set of multiple-choice questions about the main topic introduced in the lesson. Figure 3 shows the interface of the Kahoot! environment. It contains an example question proposed by the teacher in the classroom (on the left-hand side of the image) and the interface used by the students to answer via a smartphone (on the right-hand side).

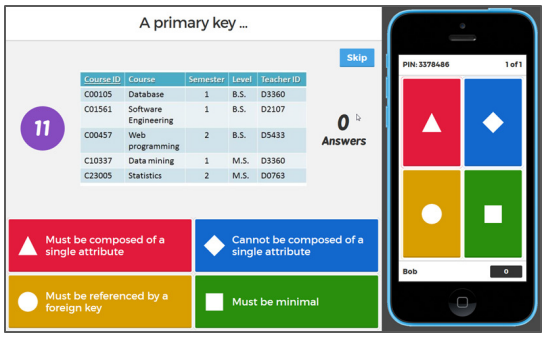

FIGURE 3. Mobile formative assessment. The Kahoot! interface.

Table 1 summarizes the main characteristics of the performed tests.

The main purposes of this activity are (i) Analyzing the level of understanding of the topics taught in the frontal lectures, and (ii) Assessing the quality of the personalized summary recommendations generated from the textbook [29].

The discussion on each of these objectives, and the connection between them, will follow in the next sections.

A. STUDENTS' PERFORMANCE ANALYSIS

The tests proposed to the students are relative to the five major topics covered by the database course (see Table 1). The tests consisted of a variable number of questions, and for each of them we collected all the students' answers, with the objective of understanding the students' attention level and the level of comprehension of the topic. The questions, in fact, are relative to what has been explained in the current lecture and not on previously covered topics. The number of students in the classroom is quite high, i.e. about 120-140 people, and this situation often results in a sub-optimal attention level. On average, approximately one third of the students present in the classroom participated to the proposed test activities (40-50 students depending on the test). The following couple sample multiple-choice tests refer to a representative topic, i.e. *relational model*. The correct answers are written in boldface, while the number of students who selected each answer is indicated in brackets. Since some of the students did not select an answer within the maximum allowed time,

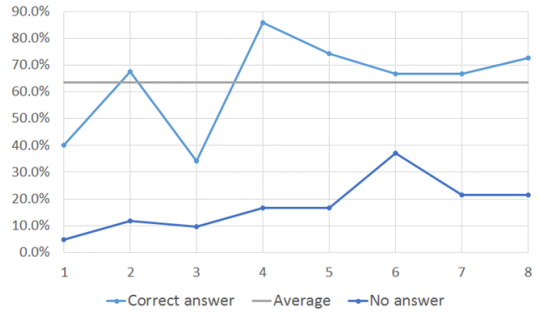

FIGURE 4. Topic: relational model: percentage of correct answers and of not given answers for each question of the test.

the number of answers is sometimes lower than the number of participants.

Question 1: What is the cardinality of a relation?

- *Choice (a): The cardinality is the set of attributes in the relation* (6 students)
- *Choice (b): The cardinality is the number of attributes in the relation* (10 students)
- *Choice (c): The cardinality is the set of n-tuples in the relation* (8 students)
- **Choice (d): The cardinality is the number of n-tuples in the relation** (10 students)

Question 5: Which of the following sentences, about primary key, is correct?

- *Choice (a): Primary keys must be composed of a single attribute* (5 students)
- *Choice (b): Primary keys cannot be composed of a single attribute* (0 students)
- *Choice (c): Primary keys must be referenced by a foreign key* (4 students)
- **Choice (d): Primary keys must be minimal** (26 students)

Figure 4 shows, for each of the eight test questions related to topic *relational model*, the percentages of correct and not given answers. In the graph the average percentage of the correct answers is also indicated (horizontal line). This graph allows the teacher to identify the concepts that are less clear to students, to provide specific reinforcement (e.g. the difference between the schema and the instance of a relation, in the given example).

The teacher can also extract useful information from the most frequent wrong answers, because often they are the symptom of a common misunderstanding. Figure 5 shows the result of such an analysis for the relational model test. The graph shows, for each question, the relative frequency of the three wrong answers. When one of the three answers is much more frequent than the others (e.g. question number 3, where most of the students selected the second answer) the teacher can better tailor the specific reinforcement.

Collected data are anonymous, but inside a single test session we could extract the general performance of the students.

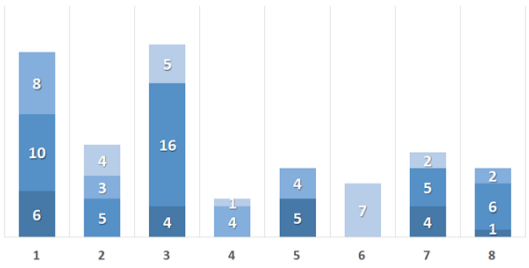

FIGURE 5. Test on relational model: relative frequency of wrong answers.

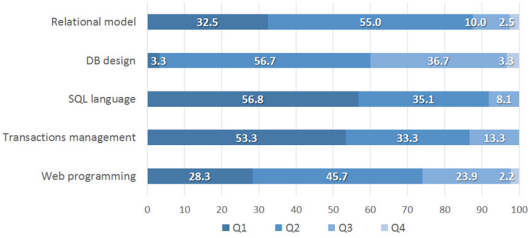

FIGURE 6. Comparison of students' performance categories in the different topics.

We divided students into four categories, according to the number of correct answers: Q1 (more than 75% of correct answers), Q2 (between 50% and 75%), Q3 (between 25% and 50%) and Q4 (less than 25%). For instance, in the relational model topic, more than one quarter of the students (32%) belong to category Q1, 55% belong to category Q2, 10% to category Q3, and very few students belong to Q4 (3%). The picture gives an idea of the general level of attention and comprehension. Since such data are more useful when compared to analogous data about different topics, in Figure 6 we compared students' test performance in all the identified topics. The graph shows very clearly the topics for which the teacher's explanation was more effective (e.g. *SQL language* and *Transactions management*, where more than half of the students belong to Q1) and the ones that left many doubts to the students (e.g. *DB design*, where very few students caught all the concepts at the first time). The graph reported in figure 7 is also useful for comparing the comprehension level of the students. The area shows the average percentage of wrong answers per student (e.g. on average students answered incorrectly to 24.1% of questions about transactions management, but to 52% of questions on DB design). The bars show instead the percentage of non-active users in each test topic, i.e. the students that registered to the test but did not give answers to more than 50% of the questions. The comparison shows once again that topics like transactions management and SQL language created less problems than DB design or Web programming. The results of this analysis helped us in defining a strategy we explained in Section III-D, on how to better exploit the extracted summaries.

B. ANALYSIS OF THE GENERATED SUMMARIES

We summarized the reference textbook of the database course. The text of the book, excluding tables, figures,

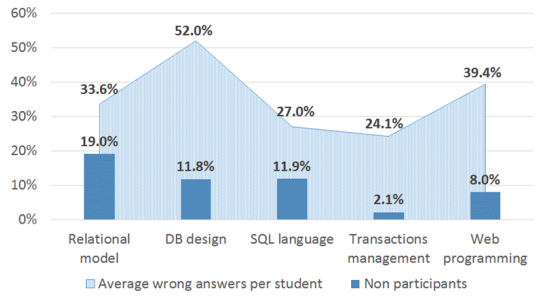

FIGURE 7. Average percentage of wrong answer per student, and number of students that did not participate actively to the test, topic by topic.

citations, and indices, was given in input to the summarization algorithm. To take the structure of the book into account during the summarization process, each chapter of the book was considered as a separate document. Note that since each chapter covers a different topic, the summarizer is able to discriminate among salient concepts related to different topics. To run the summarization algorithm [8] we used the default configuration setting for English-written documents (Wminsup = 0.8%). We generated summaries related to (i) all the general topics covered by each of the five evaluation sessions (i.e., *relational model, DB design, SQL language, transaction management, Web programming*), and (ii) a subset of specific topics covered by each test, separately for questions and answers. According to the notation introduced in Section III-D, summaries of category (i) will be hereafter denoted as *topic summaries*. They summarize the most salient content of a broad topic thus highlighting the key aspects that learners should deepen in their study or review. To extract topic summaries we exploited the on-topic dictionaries populated from the multiple-choice tests (see Section III-D).

Summaries belonging to category (ii) will be generated by exploiting more targeted dictionaries, whose contained word stems occurred only in a specific question or answer. They will be denoted as *question summaries* or *answer summaries*, respectively. Question/answer summaries give more insights into a specific aspect of the general topic. As discussed in Section III-D, they can be useful for giving punctual clarifications.

To populate on-topic dictionaries, we applied the text preparation steps described in Section III-B on the both questions and answers. The penalty score was set to 0.3 (meaning that the occurrences of word stems occurring in the dictionary are awarded, on average, by 70%).

Word stems with low relevance score in the document set are pruned, because they are less likely to represent interesting information. In particular, we considered only the word stems in the first two quartiles according to the distribution of the term relevance score in the test set. The resulting set of word stems has been validated by a domain expert prior to running the summarization process to prune irrelevant words, which accidentally occurred in the dictionaries. In our tests, approximately between 10% and 15% of the word stems selected by the procedure described above were filtered out, because they represent redundant or out-of-topic information. Notice that automated topic detection algorithms (e.g. [30], [31]) can be integrated in the proposed methodology into the text preparation phase to avoid manual result exploration. However, since on-topic dictionaries are typically small (they contain from 10 to 30 word stems) they can be easily explored by domain experts through manual inspection. Furthermore, validated dictionaries can be reused to summarize multiple document sets acquired from different sources or collected in different periods. The dictionaries used in our experiments for topic summary generation are summarized in the right-hand column of Table 1, where, for the sake of readability, we reported the entire words instead of the corresponding stems.

1) EXAMPLES OF SUMMARIES

For three representative topics (i.e., *relational model*, *transactions management*, *DB design*) Table 2 reports the top-5 sentences of the corresponding topic summary (ranked by decreasing level of significance). Let us consider, for instance, *Transactions management* test with id 2. The first sentence enumerates the four well-known transaction properties: Atomicity, Consistency, Isolation, and Durability, which are usually denoted by their acronym (ACID). The sentence was selected by the automatic summarization system because it contains a combination of words included in the dictionary (i.e., a combination of word stems with high relevance score). The other selected sentences introduce the concepts of transaction and consistency (sentences 2-3 and 5, respectively), while sentence 4 clarifies the effect of a rollback operation. The summary can be useful to learners for recalling basic concepts related to transactions in the relational model while avoiding perusing the entire textbook. This summary can be useful for students who failed the test with id 2.

Let us consider now Question 5 of topic *Relational model* (test with id 1). It asked for the meaning of the primary key in the relational model (see Section IV-A). We tried to generate the relative question summary to recommend ad hoc summaries to students who gave wrong answers to this question. The result is the following:

(1) *None of the attributes of a primary key can assume the null value; thus, the definition of primary key implies an implicit definition not null for all the attributes of the primary key.*

(2) *In practice, we adopt a simple solution, which makes it possible to guarantee the unambiguous identification of each tuple and refer to it from within other relations: null values are forbidden on one of the keys (called the primary key) and usually (that is, unless specified otherwise) allowed on the others.*

(3) *The primary key constraint can be directly defined on a single attribute, or it can be defined by listing the several attributes that make up the primary key.*

Sentences (1) and (3) clarify that the primary key may consist of one or more attributes. The second sentence indicates that attributes in the primary key must take non-null

TABLE 2. Topic summaries.

Rank	Sentence	Selected by domain experts (YES/NO)
	TEST SET 1 **Relational model**	
1	In a database there is a part that is invariant in time, called the schema of the database, made up of the characteristics of the data, and a part that changes with time, called the instance or state of the database, made up of the actual values.	Yes
2	However, a single relation is not usually sufficient for this purpose: a database is generally made up of several relations, whose tuples contain common values where this is necessary in order to establish correspondences.	No
3	For this purpose, the concept of integrity constraint was introduced, as a property that must be satisfied by all correct database instances.	Yes
4	The database shows one of the fundamental characteristics of the relational model, which is often expressed by saying that it is 'value-based': the references between data in different relations are represented by means of the values of the domains that appear in the tuples.	Yes
5	In practice, we adopt a simple solution, which makes it possible to guarantee the unambiguous identification of each tuple and refer to it from within other relations: null values are forbidden on one of the keys (called the primary key) and usually (that is, unless specified otherwise) allowed on the others.	Yes
	TEST SET 2 **Transactions management**	
1	Transactions must possess particular properties: atomicity, consistency, isolation and durability.	Yes
2	A transaction identifies an elementary unit of work carried out by an application, to which we wish to allocate particular characteristics of reliability and isolation.	Yes
3	A transaction can be defined syntactically: each transaction, irrespective of the language in which it is written, is enclosed within two commands: begin transaction (abbreviated to bot) and end transaction (abbreviated to eot).	No
4	Before executing the commit of its atomic unit, any failure will cause the elimination of all the effects of the transaction, whose original state is recreated.	Yes
5	Consistency demands that the carrying out of the transaction does not violate any of the integrity constraints defined on the database.	Yes
	TEST ID 3 **DB design**	
1	Other systems provide tools to carry out the reverse operation: reconstructing a conceptual schema based on an existing relational schema.	No
2	The generalization is transformed into two one-to-one relationships that link the parent entity E with the child entities E_1 and E_2.	Yes
3	The aim of logical design is to construct a logical schema that correctly and efficiently represents all of the information described by an Entity- Relationship schema produced during the conceptual design phase.	Yes
4	Remember that entities identified externally always participate in the relationship with a minimum and maximum cardinality of one; this type of translation is valid independently of the cardinality with which the other entities participate in the relationship.	Yes
5	Normalization allows the non-normalized schemas to be transformed into new schemas for which the satisfaction of a normal form is guaranteed.	Yes

values in all the tuples and the assumed values are unique. In the generated summary, the key concepts behind the use of primary keys in the relational model are concisely expressed.

We tried also to generate the answer summaries for the answers of test with id 5. For example, for choices (a) *Primary key must be composed of a single attribute* and (b) *Primary key cannot be composed of a single attribute* the corresponding summaries have, as top ranked sentence, the third one in the question summary. The selected sentence gives a punctual clarification on a specific aspect covered by the answer. Therefore, it may help learners to overcome misconceptions.

C. SUMMARY EVALUATION

To assess the pertinence of the generated summaries, we asked the professor and the teaching assistant of the B.S. database course to underline the parts of the textbook that they judged as mostly pertinent to the topics covered by each test. Then, we compared the generated summary with the expectation to quantify the correctness and completeness of the result. More specifically, to validate the results of the summarization process we verified the presence of each sentence of the summary in the highlight textbook content (hereafter denoted as *expectation* for the sake of brevity) and we evaluated summarizer performance in terms of an established quality measure, i.e., the precision at k, which has largely been used in recommendation systems [27].

TABLE 3. Summary evaluation. Corpus size = 157.

Test id	Topic	Summary size (# sentences)	Expectation size (# sentences)	P@5 (%)	P@10 (%)
1	Relational model	12	21	80	70
2 2	Transactions management	9	14	80	80
3	DB design	14	23	80	90
4	SQL language	17	26	70	60
6	Web programming	7	11	80	60

Precision at k ($p@k$) indicates the proportion of the top-k sentences in the summary that were expected, i.e., the number of top-k summary sentences that were underlined by the domain experts divided by k.

Since in our experiments the summary size is typically between 10 and 15, we considered as reference measures $p@5$ and $p@10$, respectively. To compute these statistics, summaries were truncated to pick only the top-k most significant sentences. Note that since the summary size is not fixed a priori, but can potentially change at any algorithm execution depending on the input data distribution, we did not considered the recall measure (the percentage of expected sentences that actually occurred in the summary) because we deemed its values as less relevant in our context of analysis.

Table 3 summarizes the results of the quantitative evaluation, where we reported for each topic/test the size of the corresponding summary, the size of the set of expected

sentences, the precision at 5, and the precision at 10. Furthermore, to give more insights into the generated summaries in the right-hand side column of Table 2 we indicated whether each of the top-5 sentences was part of the expected ones or not.

The achieved results show that

- For all the considered topics the summarizer achieved a $p@10$ value above 60% and, in half of the cases, it was above or equal to 70%. Overall, we observed that the sentences in the summary were mostly pertinent to the topic under analysis and that, in most cases, they met the expectation of the domain experts.
- For four out of six topics 80% of the sentences in the top-5 list appeared in the expected summary, while for the remaining topics at least 60% of the sentences appeared. Thus, the average $p@5$ value achieved by our method was 73%. Furthermore, for most of the considered topics the three most relevant sentences appeared in the expectation.
- Broader topics (e.g. *SQL language*) achieved, on average, a lower precision than more specific ones (e.g. *transactions management*), because since the corresponding dictionaries are less specific, summaries may contain non-pertinent sentences as well.

As discussed in the next section, to further enhance summarizer performance, the experts may enrich the dictionaries with more specific terms or may tune the value of the penalty score parameter to its best value by running the summarization algorithm multiple times for each topic and document set.

D. PARAMETER ANALYSIS

We empirically evaluated the effect of the penalty score on the quality of the achieved results. Specifically, we performed several experiments on the analyzed document sets by varying the value of penalty score between zero and one. Setting values of the penalty score close to zero rewards the occurrence of the word stems of the dictionary in the analyzed documents with respect to those of other word stems. Oppositely, setting values of penalty score close to one implies considering dictionary stems as important as the others.

Figure 8 plots the $p@5$ and $p@10$ values achieved on the document set by considering a representative topic (*Transactions management*) and by setting different values for the δ parameter. $p@5$ values are slightly higher than $p@10$ values for all the tested configuration settings, because most relevant sentences are placed at the top of the summary. The best results were achieved by setting the penalty score between 0.3 and 0.4 because a good balancing between content generalization and specialization is achieved. Setting a very high δ value significantly decreases the precision of the proposed method, because the summary because too generic and not very focused. On the other hand, by setting very low δ values the summary becomes too much focused on the dictionary content to effectively summarize generic document sets.

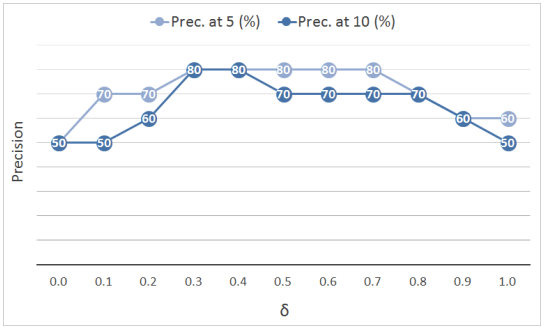

FIGURE 8. Influence of parameter δ.

V. CHALLENGES AND FUTURE PERSPECTIVES

In recent years, learning data have continuously grown due to the advances in mobile and Web-based learning technologies and the massive use of digital learning materials. Hence, one of the most appealing challenges in learning analytics is the scalability towards big datasets [1]. This prompts the need for developing scalable summarization systems for learning documents.

The scalability of existing summarization algorithms towards big document sets is limited by:

(i) The variety of languages in which learning documents are written.

(ii) The inherent complexity of data mining models, whose computational complexity is often combinatorial.

(iii) The low manageability of the data mining models, which are hardly explorable by non-expert users.

As future work, we will study scalable solutions for learning document summarization and content recommendation. Specifically, we plan to adapt and extend the existing summarization algorithm to scale towards Big learning data collections. Furthermore, we aim at integrating the recommendation and summarization steps in order to generate personalized summaries on demand. Instead of generating only textual summaries, one step further will be the analysis of the transcript of video-lectures to recommend portions of videos according to learners' needs.

VI. CONCLUSIONS

In this paper, a methodology to support the exploration of large collections of learning documents is presented. Learners are provided with short textual summaries giving a concise description of the key aspects related to specific topics. Summaries are recommended to learners according to the level of understanding of the frontal lecture. To this purpose, the outcomes of multiple-choice tests have been exploited to drive the generation and recommendation of the textual summaries. The produced recommendations range from short summaries of very broad topics (e.g., the topic covered by an entire lecture) to summaries of specific topics covered by single questions or answers in the tests.

We assessed the usability of the proposed methodology in the context of a university-level B.S. course held in our

university. The generated summaries appeared to be highly similar to the content recommended by the teacher of the course.

REFERENCES

[1] J. L. Moore, C. Dickson-Deane, and K. Galyen, "E-learning, online learning, and distance learning environments: Are they the same?" *Internet Higher Educ.*, vol. 14, no. 2, pp. 129–135, Mar. 2011.

[2] F. Ali and A. George, "Impact of a formative e-assessment on learning outcomes: A pilot study on a social and behavioural sciences course, college of health sciences, university of bahrain," in *Proc. 5th Int. Conf. e-Learn. (econf)*, Oct. 2015, pp. 408–412.

[3] M. Caeiro-Rodríguez, M. Llamas-Nistal, and F. Mikic-Fonte, "Introducing BeA into self-regulated learning to provide formative assessment support," in *Proc. IEEE Frontiers Educ. Conf. (FIE)*, Oct. 2016, pp. 1–4.

[4] L. Zhang, "Formative assessment in English for specific purposes," in *Proc. 8th Int. Conf. Measuring Technol. Mechatron. Automat. (ICMTMA)*, Mar. 2016, pp. 315–318.

[5] R. Ferguson, "Learning analytics: Drivers, developments and challenges," *Int. J. Technol. Enhanced Learn.*, vol. 4, nos. 5–6, pp. 304–317, Jan. 2012.

[6] M. S. Hossain, M. Masud, A. A. Alelaiwi, and A. Alghamdhi, "Aco-based media content adaptation for e-learning environments," in *Proc. IEEE Int. Conf. Comput. Intell. Virtual Environ. Meas. Syst. Appl.*, May 2014, pp. 118–123.

[7] S. N. Elliott, *Educational Psychology: Effective Teaching, Effective Learning*. New York, NY, USA: McGraw-Hill, 2000.

[8] E. Baralis, L. Cagliero, A. Fiori, and P. Garza, "MWI-Sum: A multilingual summarizer based on frequent weighted itemsets," *ACM Trans. Inf. Syst.*, vol. 34, no. 1, Sep. 2015, Art. no. 5.

[9] G. Siemens, "Learning analytics: Envisioning a research discipline and a domain of practice," in *Proc. 2nd Int. Conf. Learn. Anal. Knowl. (LAK)*, New York, NY, USA, 2012, pp. 4–8.

[10] A. M. Shahiri, W. Husain, and N. A. Rashid, "A review on predicting student's performance using data mining techniques," *Procedia Comput. Sci.*, vol. 72, pp. 414–422, Dec. 2015.

[11] A. Wolff, Z. Zdrahal, A. Nikolov, and M. Pantucek, "Improving retention: Predicting at-risk students by analysing clicking behaviour in a virtual learning environment," in *Proc. 3rd Int. Conf. Learn. Anal. Knowl. (LAK)*, New York, NY, USA, 2013, pp. 145–149.

[12] M. Agaoglu, "Predicting instructor performance using data mining techniques in higher education," *IEEE Access*, vol. 4, pp. 2379–2387, 2016.

[13] A. Ezen-Can, K. E. Boyer, S. Kellogg, and S. Booth, "Unsupervised modeling for understanding MOOC discussion forums: A learning analytics approach," in *Proc. 5th Int. Conf. Learn. Anal. Knowl. (LAK)*, New York, NY, USA, 2015, pp. 146–150.

[14] I. G. Rojas and R. M. C. García, "Towards efficient provision of feedback supported by learning analytics," in *Proc. IEEE 12th Int. Conf. Adv. Learn. Technol. (ICALT)*, Washington, DC, USA: IEEE Computer Society, 2012, pp. 599–603.

[15] X. Zhou, B. Wu, and Q. Jin, "Open learning platform based on personal and social analytics for individualized learning support," in *Proc. IEEE 12th Intl. Conf. Ubiquitous Intell. Comput.*, Aug. 2015, pp. 1741–1745.

[16] A. J. Stimpson and M. L. Cummings, "Assessing intervention timing in computer-based education using machine learning algorithms," *IEEE Access*, vol. 2, pp. 78–87, 2014.

[17] E. Baralis and L. Cagliero, "Learning from summaries: Supporting e-learning activities by means of document summarization," *IEEE Trans. Emerg. Topics Comput.*, vol. 4, no. 3, pp. 416–428, Jul./Sep. 2016.

[18] L. Cagliero, L. Farinetti, and E. Baralis, "Test-driven summarization: Combining formative assessment with teaching document summarization," in *Proc. IEEE 41st Annu. Comput. Softw. Appl. Conf. (COMPSAC)*, Turin, Italy, Jul. 2017, pp. 401–406. doi. 10.1109/COMPSAC.2017.231.

[19] G. Yang, D. Wen, Kinshuk, N.-S. Chen, and E. Sutinen, "Personalized text content summarizer for mobile learning: An automatic text summarization system with relevance based language model," in *Proc. IEEE 4th Int. Conf. Technol. Educ. (T4E)*, Jul. 2012, pp. 90–97.

[20] S. Saraswathi, M. Hemamalini, S. Janani, and V. Priyadharshini, "Multi-document text summarization in E-learning system for operating system domain," in *Advances in Computing and Communications* (Communications in Computer and Information Science), vol. 193. Berlin, Germany: Springer, 2011, pp. 175–186.

[21] E. Baralis and A. Fiori, "Summarizing biological literature with biosumm," in *Proc. 19th ACM Int. Conf. Inf. Knowl. Manage. (CIKM)* 2010, Toronto, ON, Canada, Oct. 2010, pp. 1961–1962. doi: 10.1145/1871437.1871785.

[22] D. Wang, S. Zhu, T. Li, Y. Chi, and Y. Gong, "Integrating document clustering and multidocument summarization," *ACM Trans. Knowl. Discovery Data*, vol. 5, no. 3, pp. 14:1–14:26, Aug. 2011.

[23] E. Baralis, L. Cagliero, S. Jabeen, A. Fiori, and S. Shah, "Multi-document summarization based on the Yago ontology," *Expert Syst. Appl.*, vol. 40, no. 17, pp. 6976–6984, Dec. 2013.

[24] M. Dredze, H. M. Wallach, D. Puller, and F. Pereira, "Generating summary keywords for emails using topics," in *Proc. 13th Int. Conf. Intell. User Interfaces (IUI)*, New York, NY, USA, 2008, pp. 199–206.

[25] C.-Y. Lin and E. Hovy, "Automatic evaluation of summaries using N-Gram co-occurrence statistics," in *Proc. North Amer. Chapter Assoc. Comput. Linguistics Human Language Technol.*, vol. 1, 2003, pp. 71–78.

[26] Y. Song, Z. Hu, Y. Guo, and E. F. Gehringer, "An experiment with separate formative and summative rubrics in educational peer assessment," in *Proc. IEEE Frontiers Educ. Conf. (FIE)*, Oct. 2016, pp. 1–7.

[27] P.-N. Tan, M. Steinbach, and V. Kumar, *Introduction to Data Mining*, 1st ed. Boston, MA, USA: Addison-Wesley, 2005.

[28] R. Agrawal, T. Imieliński, and Swami, "Mining association rules between sets of items in large databases," in *Proc. ACM SIGMOD*, 1993, pp. 207–216.

[29] P. Atzeni, S. Ceri, S. Paraboschi, and R. Torlone, *Database Systems—Concepts, Languages and Architectures*. New York, NY, USA: McGraw-Hill, 1999.

[30] Y. Chen, H. Amiri, Z. Li, and T.-S. Chua, "Emerging topic detection for organizations from microblogs," in *Proc. 36th Int. ACM SIGIR Conf. Res. Develop. Inf. Retr. (SIGIR)*, New York, NY, USA, 2013, pp. 43–52. doi: 10.1145/2484028.2484057.

[31] H. Lin, B. Sun, J. Wu, and H. Xiong, "Topic detection from short text: A term-based consensus clustering method," in *Proc. 13th Int. Conf. Service Syst. Service Manage. (ICSSSM)*, Jun. 2016, pp. 1–6.

LUCA CAGLIERO received the master's degree in computer and communication networks and the Ph.D. degree in computer engineering from the Politecnico di Torino, where he has been an Assistant Professor with the Dipartimento di Automatica e Informatica, since 2016. His current research interests are in the fields of data mining and textual data analytics, specifically text summarization, classification, and association rule mining.

LAURA FARINETTI received the master's degree in electronic engineering and the Ph.D. degree in cognitive science from the Politecnico di Torino, where she has been a Senior Researcher and an Assistant Professor with the Dipartimento di Automatica e Informatica, since 2001. Her current research interests include ICT technologies in open education, learning analytics, human computer interaction, and semantic web.

ELENA BARALIS received the master's degree in electrical engineering and the Ph.D. degree in computer engineering from the Politecnico di Torino, where she has been a Full Professor with the Dipartimento di Automatica e Informatica, since 2005. She has published over 150 papers in international journals and conference proceedings. Her current research interest includes data mining, specifically on mining algorithms for big data and stream data analysis.